Predicting and Controlling Resource Usage in a Heterogeneous Active Network

Virginie Galtier, Yannick Carlinet and Kevin L. Mills (NIST)
Stephen F. Bush and Amit B. Kulkarni (GE CRD)

Center for Satellite and Hybrid Communication Networks'
Advanced Networks Colloquium

March 30, 2001

Go Terps! Beat Duke!

NIST
National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

UNIVERSITY OF MARYLAND

George Mason University

GSU

Georgetown UNIVERSITY

Stanford University

Just Ahead

- ***Advertising 101***: provide some information about an PC2001, an upcoming Pervasive-Computing conference at NIST
- ***Advertising 102***: briefly introduce NIST, the Information Technology Laboratory (ITL), and myself
- ***And Now Back to Our Irregularly Scheduled Program***
  - Why do we need to model CPU time requirements for mobile code?
  - How can we model CPU time requirements? And How Well?
  - Does this work when applied to applications?
    - Application 1: Controlling Execution of Mobile Code
    - Application 2: Predicting CPU Usage by Mobile Code

http://www.nist.gov/pc2001/

NIST National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

Featured Speakers Include:

Dr. Ambuj Goyal
Vice President,
Strategy and Architecture
IBM Corporation

Dr. Roy Want
Intel Research

Dr. Steven Shafer
Microsoft Research

Dr. K. Venkatesh Prasad
Ford Motor Company

Topics include

Applications: automotive applications, smart homes and work environments, traveler services, wireless ticketing, and mobile commerce

Technologies: multi-hop wireless networking, ad hoc networking/PANs. pico-cellular wireless, service discovery

NIST. . . working with industry to develop and apply technology, measurements, and standards.

- Advanced Technology Program
- Baldrige Quality Program
- Manufacturing Extension Partnership
- Measurement and Standards Laboratories
  - Electronics and Electrical Engineering
  - Manufacturing Engineering
  - Chemical Science and Technology
  - Physics (the all-stars!)
  - Materials Science and Engineering
  - Building and Fire Research
  - Information Technology

A few words.

Nobel Prize

I work here.

ITL . . . the Nation's information technology standards, testing, and measurement laboratory

Working with Industry

- Networking Research
- Security
- Information Access
- Software Testing
- Convergent Information Systems

I work here.

Serving NIST

- Mathematics, Statistics, and Computational Sciences
- Information Technology Services

Latest Highlight: Advanced Encryption Standard

MY CURRENT RESEARCH PROJECTS

- Predicting and controlling research usage in heterogeneous active networks - ***joint work with Virginie Galtier and Yannick Carlinet (NIST) and Steve Bush and Amit Kulkarni (GE CRD)*** *with funding from DARPA and NIST*

- Analyzing the properties and behavior of emerging service discovery protocols - ***joint work with Christopher Dabrowski (NIST)*** *with funding from ARDA, DARPA, and NIST*

- Designing and evaluating self-adaptive discovery mechanisms for optimal performance in fault-tolerant networks - ***joint work with Oliver Mathieu, Doug Montgomery, and Scott Rose (NIST)*** *with funding from ARDA, DARPA, and NIST*

- Exploring collective dynamics of large-scale networks - ***joint work with Jian Yuan and Doug Montgomery (NIST)*** *with funding from DARPA and NIST*

Predicting and Controlling Resource Usage in a Heterogeneous Active Network

Virginie Galtier, Yannick Carlinet and Kevin L. Mills (NIST)
Stephen F. Bush and Amit B. Kulkarni (GE CRD)

Center for Satellite and Hybrid Communication Networks'
Advanced Networks Colloquium

March 30, 2001

Outline of Presentation

- What is the problem? Why is it important?
- How do we try to solve the problem?
  - Modeling CPU-Time Use by Mobile Code
  - Scaling Our Models Among Heterogeneous Nodes
- Is our solution good for anything?
  - Application #1: Control Execution of Mobile Code
  - Application #2: Predict CPU Consumption among Heterogeneous Nodes in a Network
- What's wrong with our current solution?

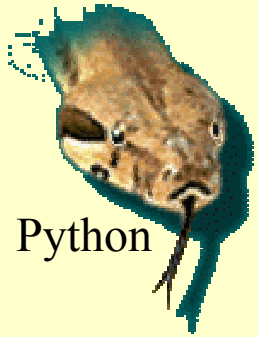
What's the Problem Anyway?

Why Is It Important?

Who Might Care?

Growing Population of Mobile Programs on Heterogeneous Platforms

SCRIPTING ENGINES & LANGUAGES



vbscript
jscript

APPLETS & SERVLETS

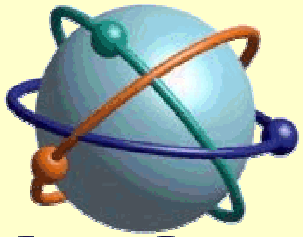
ACTIVE NETWORKS

dlls, dlls, and more dlls

Microsoft

C#

MOBILE AGENTS

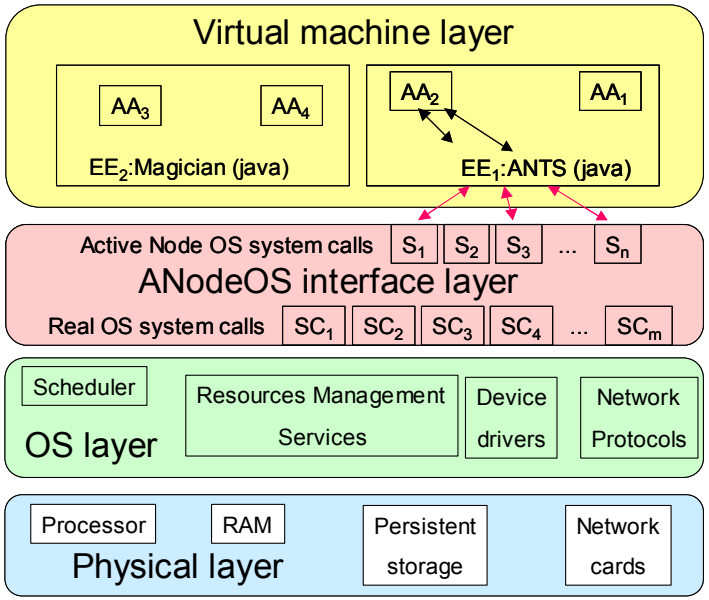


How Do We Model an Application's CPU-Time Usage?

How Well Do the Models Match Reality?

How Do We Know?

Sources of Variability



ANETS ARCHITECTURE

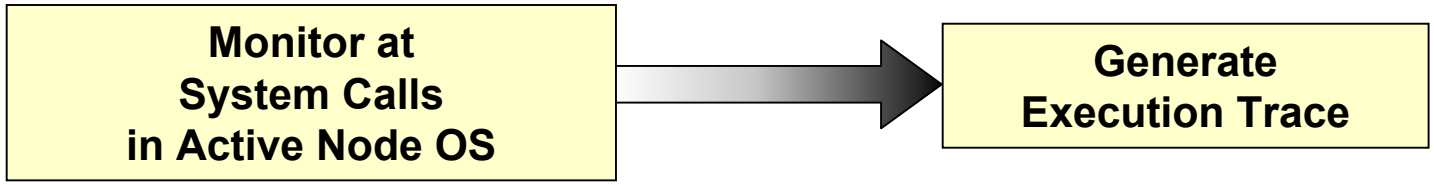
VARIABILITY IN EXECUTION ENVIRONMENT

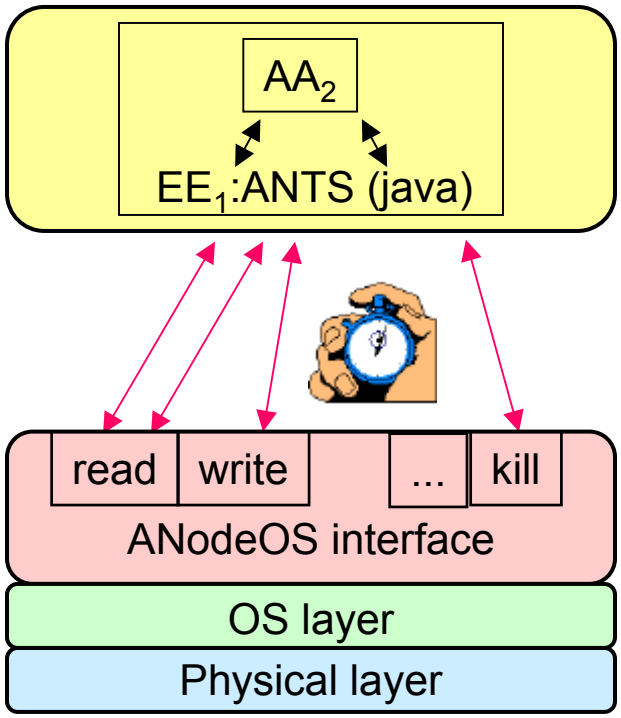
Trait	Blue	Black	Green
CPU Speed	450 MHz	333 MHz	199 MHz
Processor	Pentium II	Pentium II	PentiumPro
Memory	128 MB	128 MB	64 MB
OS	Linux 2.2.7	Linux 2.2.7	Linux 2.2.7
JVM	jdk 1.1.6	jdk 1.1.6	jdk 1.1.6
Benchmark			
Avg. CPU us	534	479	843
Avg. PCCs	240,269	159,412	167,830

	Blue		Black		Green	
System Call	pcc	us	pcc	us	pcc	us
read	19,321	43	12,362	37	12,606	63
write	22,609	50	14,394	43	12,362	62
socketcall	27,066	60	17,591	53	14,560	73
stat	22,800	51	14,731	44	12,042	61

VARIABILITY IN SYSTEM CALLS

Measuring AA Executions





…
begin, user (4 cc), read (20 cc), user (18 cc), write(56 cc), user (5 cc), end

begin, user (2 cc), read (21 cc), user (18 cc), kill (6 cc), user (8 cc), end

begin, user (2 cc), read (15 cc), user (8 cc), kill (5 cc), user (9 cc), end

begin, user (5 cc), read (20 cc), user (18 cc), write(53 cc), user (5 cc), end

begin, user (2 cc), read (18 cc), user (17 cc), kill (20 cc), user (8 cc), end
…

Trace is a series of system calls and transitions stamped with CPU time use

Modeling AA Executions

Consume Execution Trace → **Generate Active Application Model**

…
begin, user (4 cc), read (20 cc), user (18 cc), write(56 cc), user (5 cc), end

begin, user (2 cc), read (21 cc), user (18 cc), kill (6 cc), user (8 cc), end

begin, user (2 cc), read (15 cc), user (8 cc), kill (5 cc), user (9 cc), end

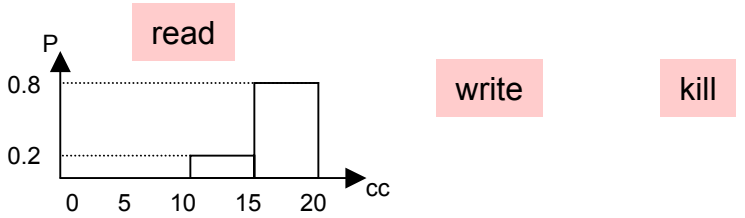
begin, user (5 cc), read (20 cc), user (18 cc), write(53 cc), user (5 cc), end

begin, user (2 cc), read (18 cc), user (17 cc), kill (20 cc), user (8 cc), end
…

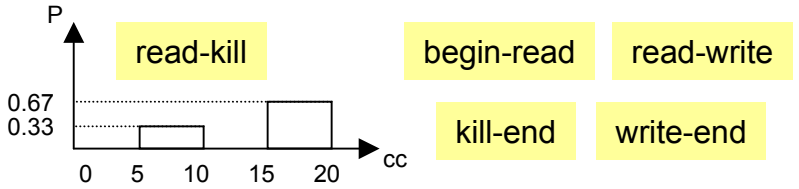
Scenario A:
sequence = "read-write",
probability = 2/5

Scenario B:
sequence = "read-kill",
probability = 3/5

Distributions of CPU time in system calls :



Distributions of CPU time between system calls :



Evaluating AA Models

Simulate Model with Monte Carlo Experiment → **Statistically Compare Simulation Results against Measured Data**

		100 bins-20000 reps		50 bins-20000 reps		50 bins-500 reps	
EE	AA	Mean	Avg. High Per.	Mean	Avg. High Per.	Mean	Avg. High Per.
ANTS	Ping	0.86	0.9	0.64	2	2.70	10
	Mcast	0.40	1.9	0.35	3	4.91	16
Magician	Ping	0.44	33	0.70	32	1.77	32
	Route	0.73	13	0.30	12	6.66	23

The Average Absolute Deviation (in Percent) of Simulated Predictions from Measured Reality for Each of Two Active Applications in Two Different Execution Environments Running on One Node (Average High Percentile Considers Combined Comparison of 80th, 85th 90th, 95th, and 99th Percentiles) –Results Given for Models Composed Using Three Different Combinations of Bin Granularity (bins) and Simulation Repetitions (reps)

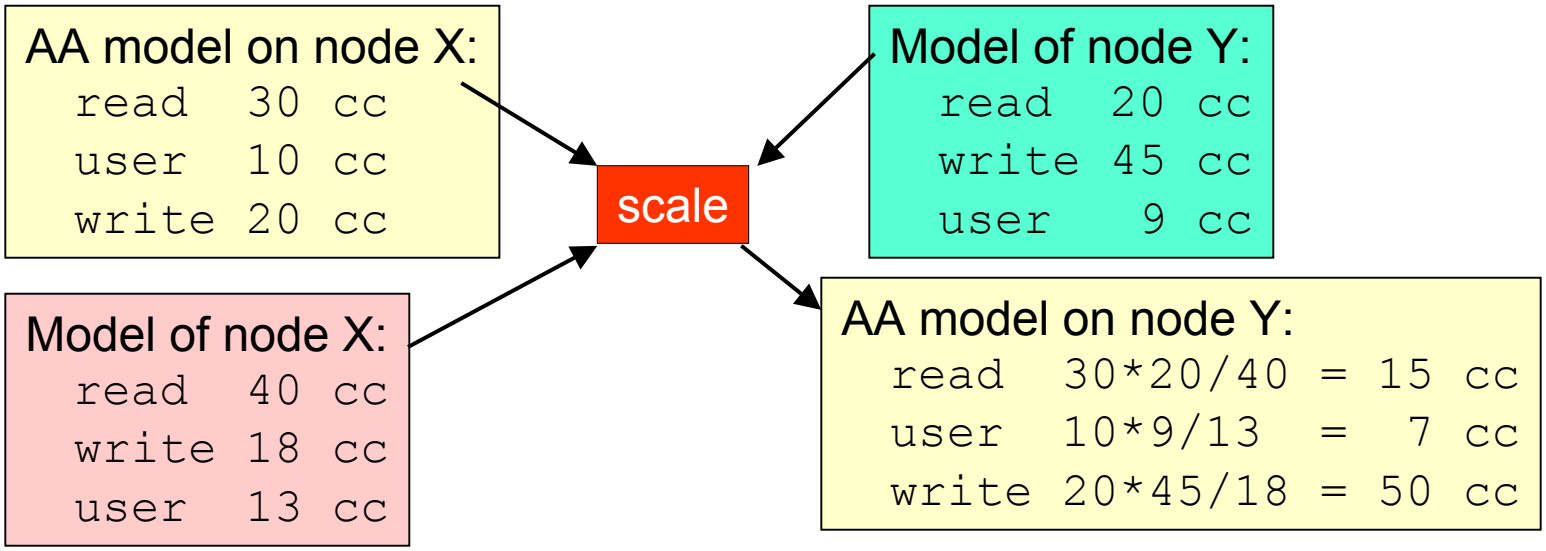
How Can We Scale Our Models For Understanding Among Heterogeneous Nodes?

How Well?

How Do We Know?

Scaling AA Models

- Each Node Constructs a Node Model using two benchmarks:
  - a system benchmark program ⌫ for each system call, average system time
  - for each EE, a user benchmark program ⌫ average time spent in the EE between system calls
- To scale an AA Model select one Node Model as a reference known by all other active nodes



Evaluating Scaled AA Models

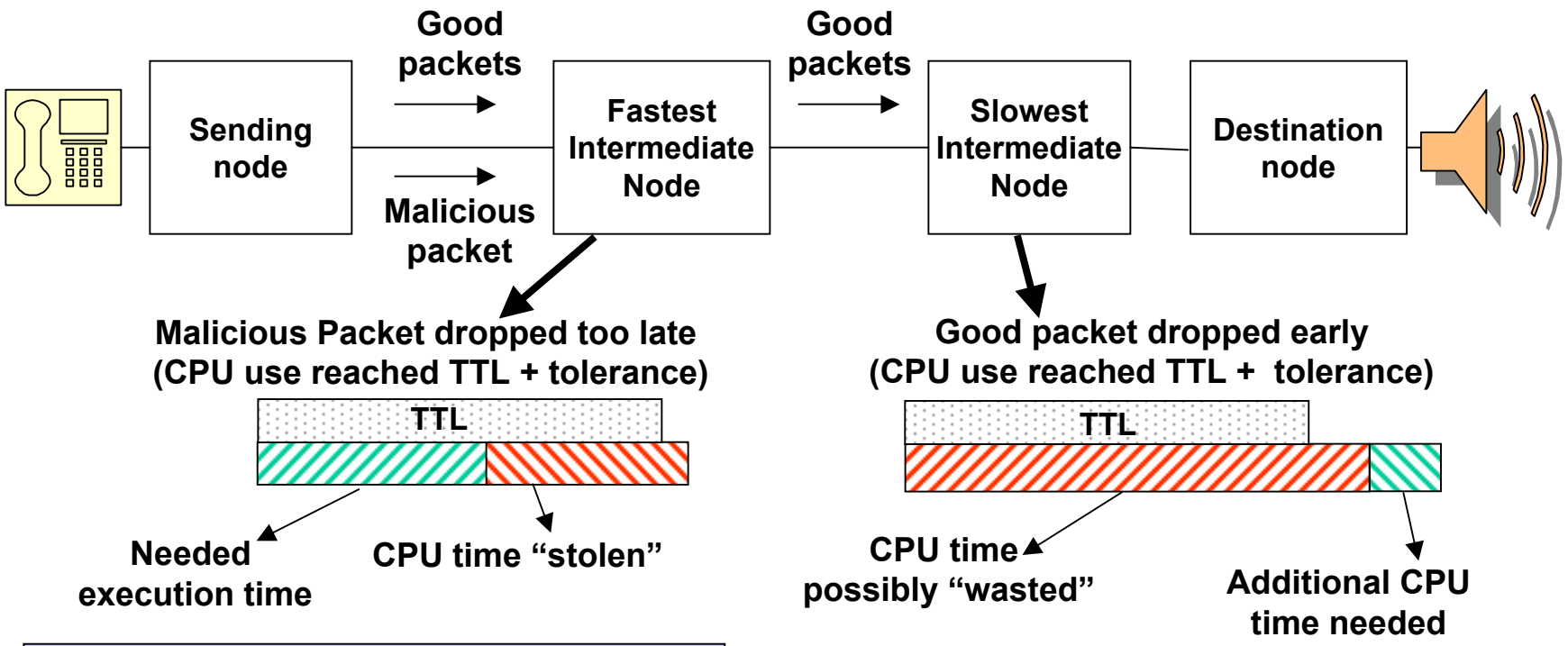
Prediction Error Measured when Scaling Application Models between Selected Pairs of Nodes vs. Scaling with Processor Speeds Alone

			Scaling with Models		Scaling with Speeds	
AA	Node X	Node Y	Mean	Avg. High Per.	Mean	Avg. High Per.
Ping	Blue	Black	<1	21	15	38
	Blue	Green	2	18	13	15
	Black	Blue	<1	16	13	25
	Red	Green	6	10	92	82
	Red	Black	4	14	154	135
	Yellow	Black	6	16	190	163
	Yellow	Green	8	15	119	103
	Black	Green	4	23	24	22
Route	Blue	Black	2	9	15	250
	Black	Blue	<1	23	13	32
	Red	Green	4	15	88	64
	Red	Black	6	19	155	137
	Yellow	Black	5	16	190	164
	Yellow	Green	6	14	114	83
	Black	Green	3	28	26	28
	Blue	Green	<1	28	15	204

Is This Good For Anything?

Application #1: Controlling Execution of Mobile Code

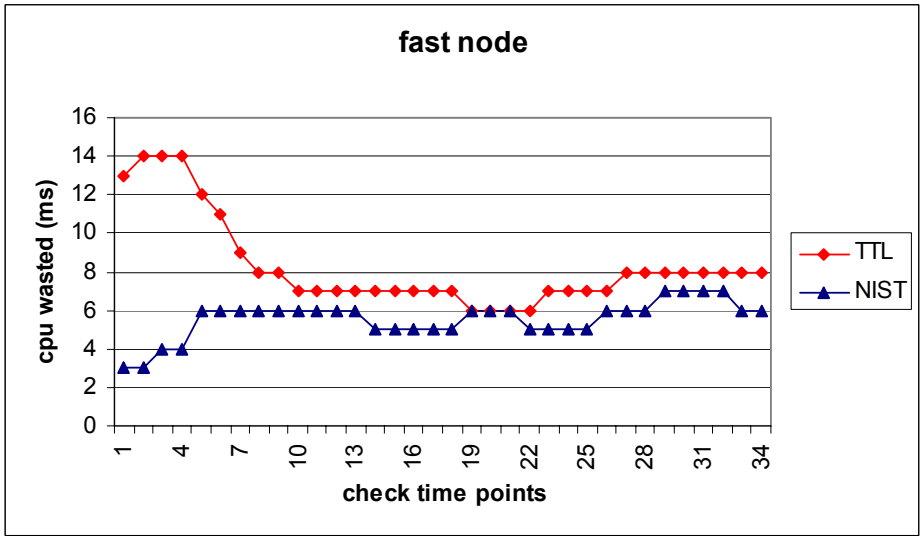
Experiment in Progress: Control CPU Usage by Mobile Programs

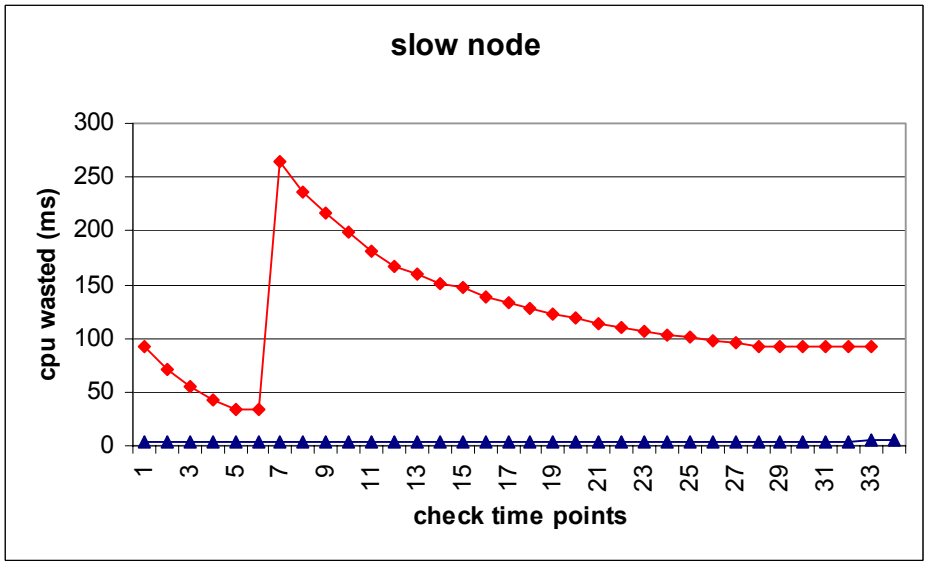


When mobile code CPU usage controlled with fixed allocation or TTL, malicious or "buggy" mobile programs can "steal" substantial CPU cycles, especially on fast nodes

When mobile code CPU usage controlled with fixed allocation or TTL, correctly coded mobile programs can be terminated too soon on slow nodes, wasting substantial CPU cycles

CPU Control: Expected Results



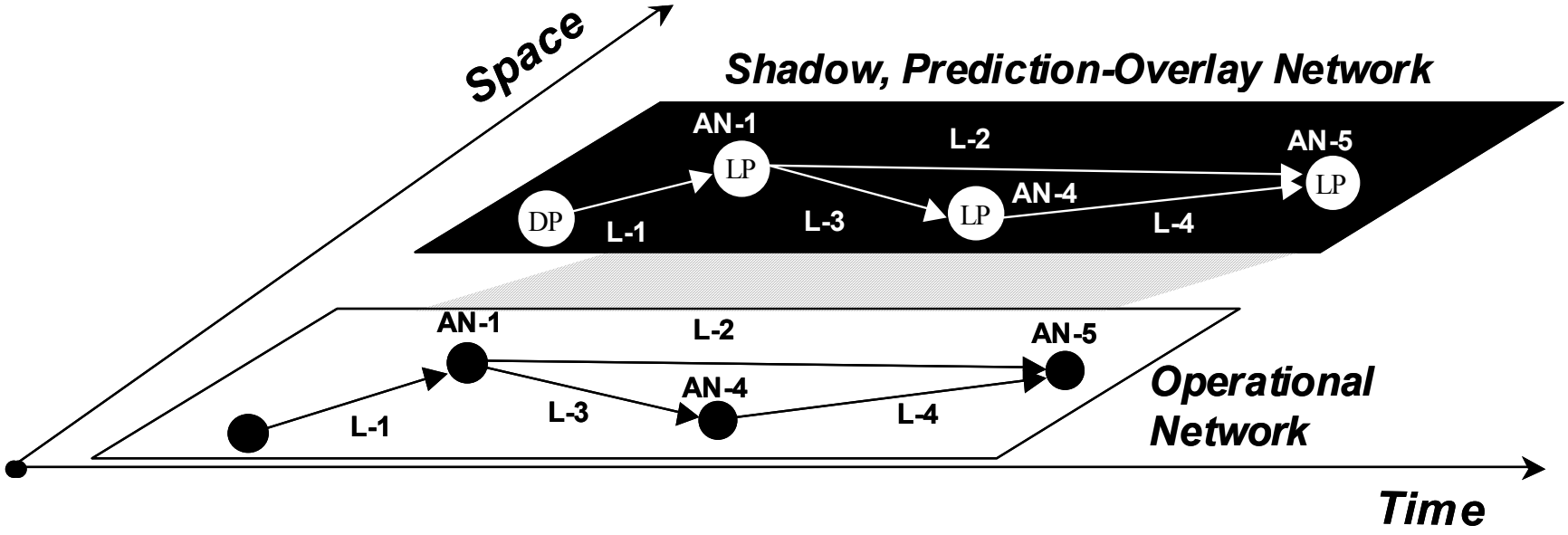


Is This Good For Anything?

Application #2:

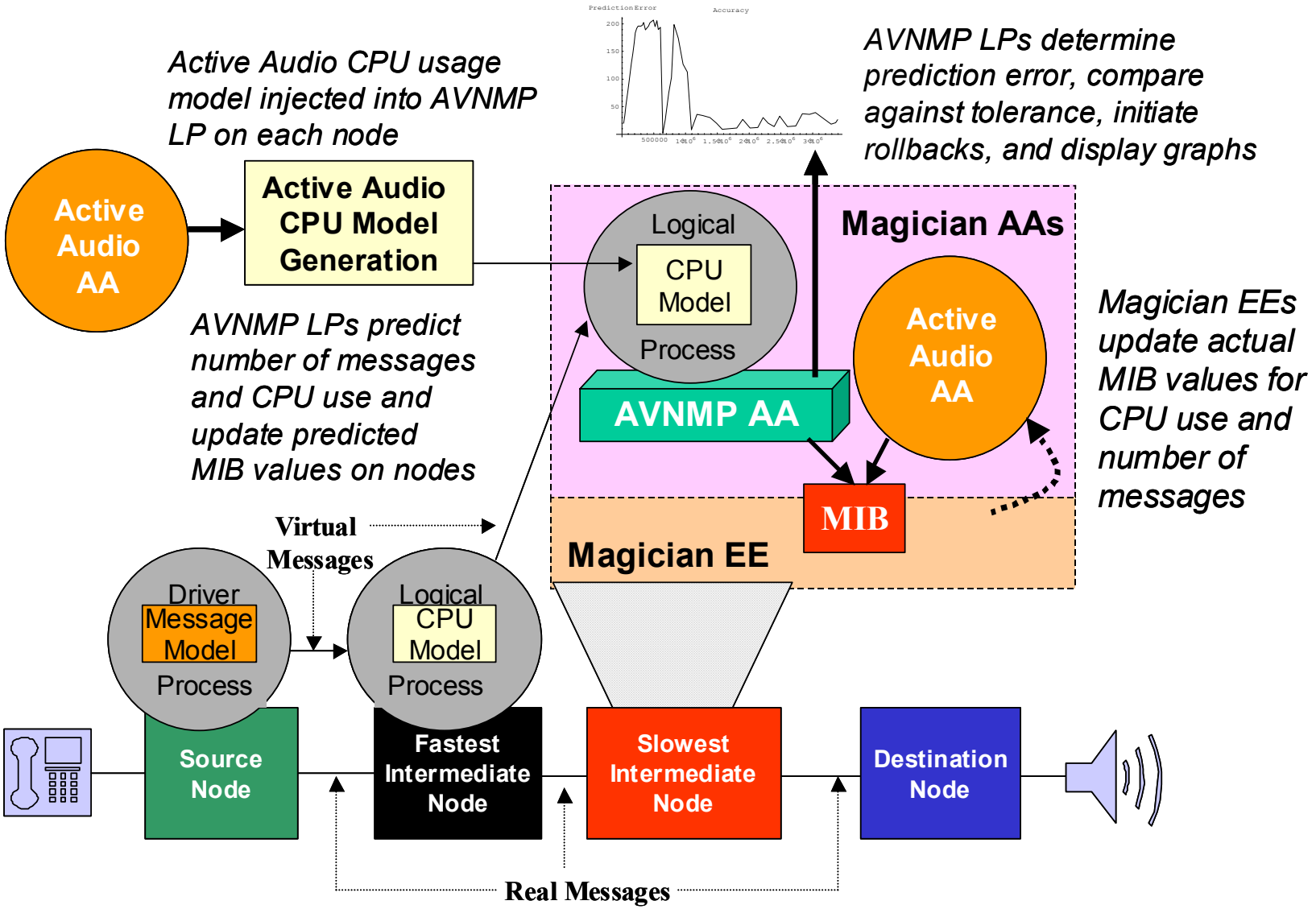
Predicting CPU Use by Mobile Code

GE Active Virtual Network Management Prediction (AVNMP) System

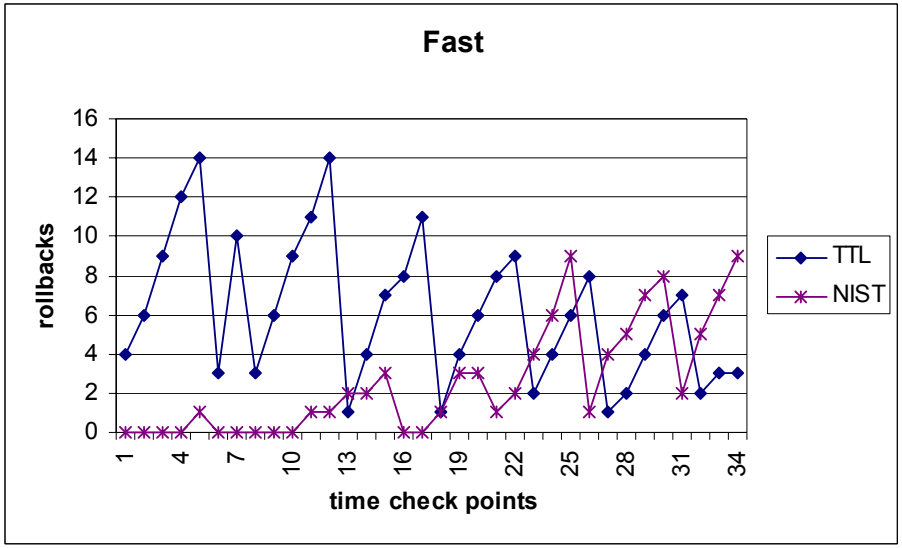


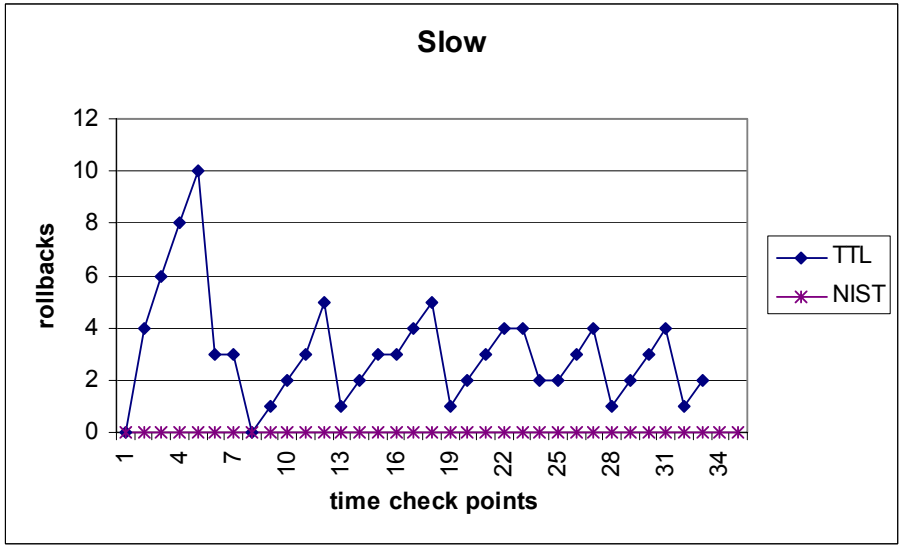
Can NIST Models enable AVNMP to predict CPU use among heterogeneous network nodes, while providing better look ahead and improved prediction efficiency than simple TTL approaches?

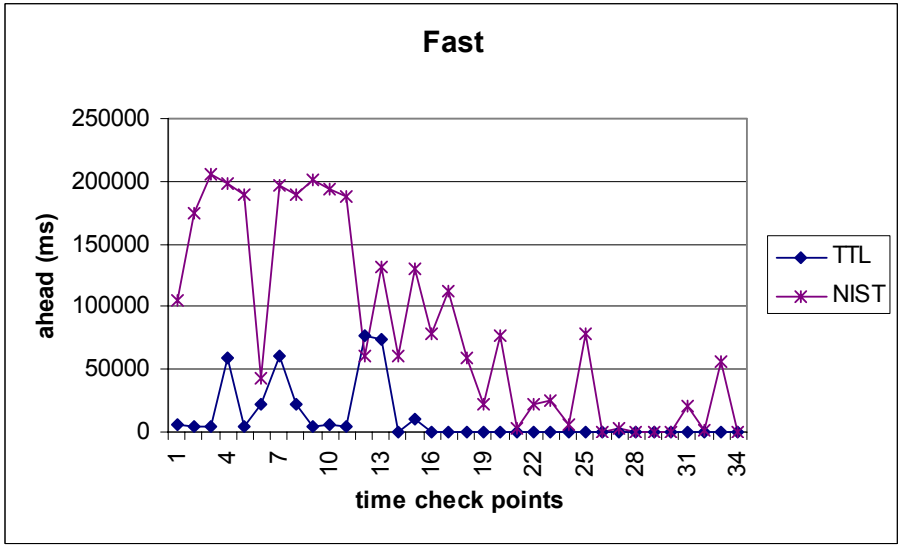
Experiment in Progress: Predict CPU Usage among Heterogeneous Network Nodes

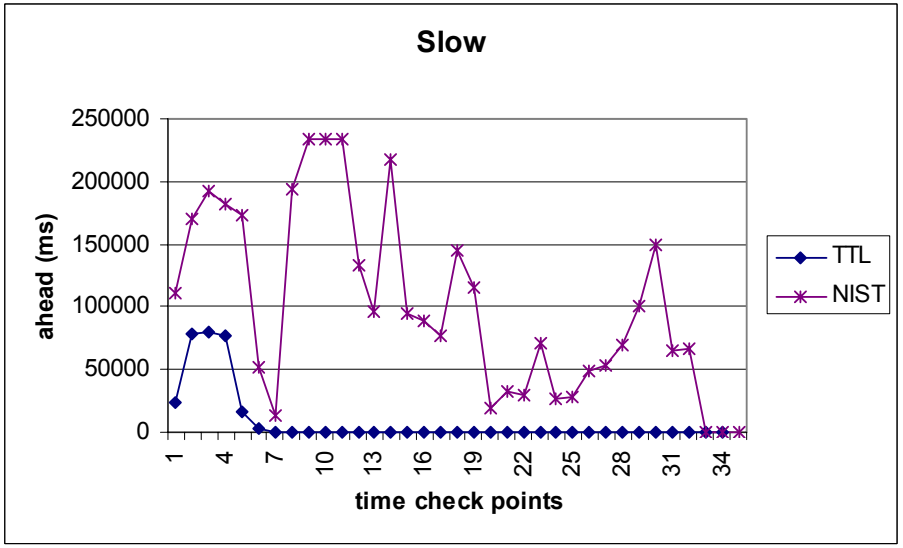


CPU Prediction: Expected Results









What's wrong with our current solution?

The Three Biggest Problems

- Need to Improve Space-Time Efficiency of Our Models
- Need to Account for Node-Dependent Conditions
- Need to Characterize Error Bounds of Our Models
- So, future research needs to address these issues

Other Future Research

- Explore Additional Applications
  - Scheduling Tasks in a Distributed Server Farm *(this one is the subject of current experiments by Yannick Carlinet)*
  - Network Path Selection Mechanisms that Consider CPU Requirements
- Investigate Alternate Models
  - White-box Models
  - Lower-Complexity Analytically Tractable Models
  - Models that Learn
- Investigate Prediction based on Competition
  - Run and Score Competing Predictors for Each Application
  - Reinforce Good Predictors
  - Use Prediction from Best Scoring Model

Some Related Publications

V. Galtier, C. Hunt, S. Leigh, K. Mills, D. Montgomery, M. Ranganathan, A. Rukhin, and D. Tang, "How Much CPU Time?", *Draft NIST Technical Report*, TR-ANTD-ANETS-111999, November 1999. <http://w3.antd.nist.gov/~mills/unpublished/NISTanetsTR.pdf>

Y. Carlinet, V. Galtier, K. Mills, S. Leigh, A. Rukhin, "Calibrating an Active Network Node," *Proceedings of the 2nd Workshop on Active Middleware Services*, ACM, August 2000. <http://w3.antd.nist.gov/~mills/papers/Final-woasm.pdf>

V. Galtier, K. Mills, Y. Carlinet, S. Leigh, A. Rukhin, "Expressing Meaningful Processing Requirements among Heterogeneous Nodes in an Active Network," *Proceedings of the 2nd International Workshop on Software Performance*, ACM, September 2000. <http://w3.antd.nist.gov/~mills/papers/WSOPfu-04.pdf>

V. Galtier, K. Mills, Y. Carlinet, S. Bush, and A. Kulkarni, "Predicting Resource Demand in Heterogeneous Active Networks", submitted to *MILCOM 2001*. <http://w3.antd.nist.gov/~mills/unpublished/ALTmilcom2001v4.pdf>

And don't forget the project web site:

http://w3.antd.nist.gov/active-nets/